

Objektno orjentisano programiranje – C++

Vežbe – Korisničke klase. Nasleđivanje



Zadatak 1

- Napisati klasu *Bankovni_racun*. Klasa ima 3 privatne promenljive *broj_racuna* tipa *string* te *stanje_racuna* i *kamatna_stopa* tipa *double*. Klasa ima:
 - javni konstruktor koji prethodnim promenljivim dodeljuje vrednosti.
 - *double* funkciju *Izracunaj_kamatu* koja vraća kamatu prema obrascu $\text{stanje_racuna} * \text{kamatna_stopa}$.

U glavnom programu:

- deklarirati promenljive za unos broja računa, stanja na računu i kamatne stope a zatim uneti te podatke.
- deklarirati objekat *racun1* klase *Bankovni_racun* sa parametrima prethodno definisanih promenljivih.
- Ispisati kamatu na novac na računu preko objekta *racun1* i funkcije *Izracunaj_kamatu*.

Zadatak 1

Unesite broj racuna: 1234

Unesite stanje na racunu: 1000

Unesite kamatnu stopu: 0.06

Kamata na novac koji se nalazi na racunu je: 60.00



Rešenje zadatka 1 (1)

```
#include <iostream>
#include<iomanip>
#include<string>
using namespace std;
```

```
class Bankovni_racun{
private:
    string broj_racuna;
    double stanje_racuna;
    double kamatna_stopa;
public:
    Bankovni_racun(string, double, double);
    double Izracunaj_kamatu();
};
```

```
Bankovni_racun::Bankovni_racun(string broj, double stanje, double kamata){
    broj_racuna=broj;
    stanje_racuna=stanje;
    kamatna_stopa=kamata;
}
```

```
double Bankovni_racun::Izracunaj_kamatu(){  
    return stanje_racuna*kamatna_stopa;  
}
```

```
int main(){  
    cout << setprecision(2)  
        << setiosflags(ios::fixed)  
        << setiosflags(ios::showpoint);  
    string racun;  
    double rcstanje;  
    double stopa;  
  
    cout << "Unesite broj racuna: ";  
    getline(cin, racun);  
    cout << "\nUnesite stanje na racunu: ";  
    cin >> rcstanje;  
    cout << "\nUnesite kamatnu stopu: ";  
    cin >> stopa;
```



```
Bankovni_racun racun1(racun, rcstanje, stopa);
```

```
    cout << "\nKamata na novac koji se nalazi na racunu je: " <<  
    racun1.Izracunaj_kamatu() << endl << endl;  
    return 0;  
}
```

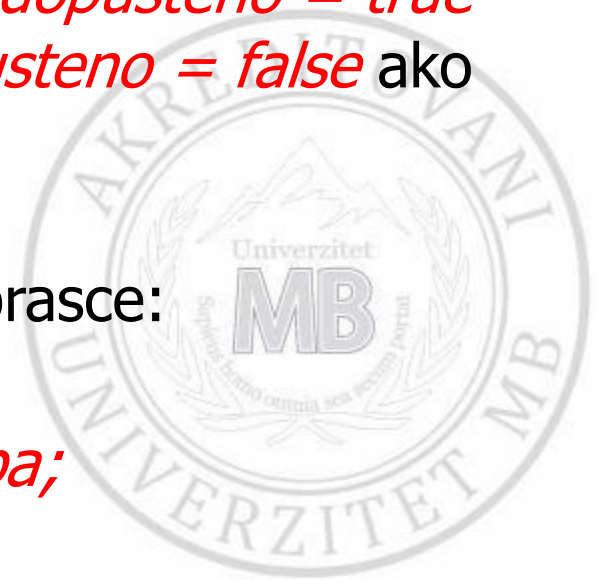


Zadatak 2

- Proširiti klasu *Bankovni_racun* iz prethodnog zadatka sa novim funkcijama:
 - *double Prosledi_stanje* koja vraća stanje na računu.
 - *void Uplata(double)* koja vraća stanje na računu prema obrascu *stanje_racuna += iznos*
 - *bool Isplata(double)* koja vraća promenljivu *dopusteno = true* ako ima dovoljno novca za podizanje a *dopusteno = false* ako nema dovoljno novca.

U funkciji *Izracunaj_kamatu* koristiti sledeće obrasce:

*kamatni_iznos=stanje_racuna*kamatna_stop;*
stanje_racuna+=kamati_iznos;



Zadatak 2

Unesite broj racuna: 1234

Unesite stanje na racunu: 100

Unesite kamatnu stopu: 0.06

Trenutno stanje na racunu je 100.00

Iznos koji ulazete: 55.42

Uplata od 55.42 je izvršena.

Novo stanje na racunu je 155.42

Primenjuje se kamatna stopa za iznos na racunu.

Novo stanje na racunu je 164.75

Iznos koji podizete: 120

Podizanje iznosa od 120.00 je izvršeno.

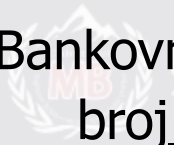
Novo stanje na racunu je 44.75

Rešenje zadatka 2 (1)

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;
```

```
class Bankovni_racun{
private:
    string broj_racuna;
    double stanje_racuna;
    double kamatna_stopa;
public:
    Bankovni_racun(string, double, double);
    double Izracunaj_kamatu();
    double Prosledi_stanje();
    void Uplata(double);
    bool Isplata(double);
};
```





```
Bankovni_racun::Bankovni_racun(string broj, double stanje, double kamata){  
    broj_racuna=broj;  
    stanje_racuna=stanje;  
    kamatna_stopa=kamata;  
}
```

```
double Bankovni_racun::Prosledi_stanje(){  
    return stanje_racuna;  
}
```

```
double Bankovni_racun::Izracunaj_kamatu(){  
    double kamatni_iznos;  
    kamatni_iznos=stanje_racuna*kamatna_stopa;  
    stanje_racuna+=kamatni_iznos;  
    return kamatni_iznos;  
}
```

```
void Bankovni_racun::Uplata(double iznos){  
    stanje_racuna+=iznos;  
}
```

Rešenje zadatka 2 (2)





```
bool Bankovni_racun::Isplata(double iznos){  
    bool dopusteno;  
    if (iznos<=stanje_racuna){  
        stanje_racuna-=iznos;  
        dopusteno=true;  
    }  
    else  
        dopusteno=false;  
    return dopusteno;  
}
```

```
int main(){  
    cout << setprecision(2)  
        << setiosflags(ios::fixed)  
        << setiosflags(ios::showpoint);  
    string racun;  
    double rcstanje;  
    double stopa;  
    double iznos;
```

Rešenje zadatka 2 (3)



```
cout << "Unesite broj racuna: ";  
getline(cin, racun);  
cout << "\nUnesite stanje na racunu: ";  
cin >> rcstanje;  
cout << "\nUnesite kamatnu stopu: ";  
cin >> stopa;
```

```
Bankovni_racun racun1(racun, rcstanje, stopa);
```

```
cout << "Trenutno stanje na racunu je " << racun1.Prosledi_stanje() <<  
endl << endl;  
cout << "Iznos koji ulazete: ";  
cin >> iznos;  
  
racun1.Uplata(iznos);  
cout << endl << endl;  
cout << "Uplata od " << iznos << " je izvršena." << endl;  
cout << "Novo stanje na racunu je " << racun1.Prosledi_stanje();
```

```
racun1.Izracunaj_kamatu();  
cout << endl << endl;  
cout << "Primenjuje se kamatna stopa za iznos na racunu." << endl;  
cout << "Novo stanje na racunu je " << racun1.Prosledi_stanje();  
  
cout << endl << endl;  
cout << "Iznos koji podizete: ";  
cin >> iznos;  
  
if(racun1.Isplata(iznos)){  
    cout << endl << endl;  
    cout << "Podizanje iznosa od " << iznos << " je izvršeno.";  
}  
else{  
    cout << endl << endl;  
    cout << "Podizanje nije izvršeno. Nemate dovoljno novca na racunu";  
}
```

Rešenje zadatka 2 (6)

```
cout << endl << endl;  
cout << endl << "Novo stanje na racunu je " << racun1.Prosledi_stanje()  
<< endl << endl;  
return 0;  
}
```



Zadatak 3

- Napisati klasu *Polygon*. Klasa ima 2 protected promenljive *sirina* i *visina* tipa *int* te javnu funkciju *void Postavi (int a, int b)* koja dodeljuje vrednosti varijablama *sirina* i *visina*.
- Napisati izvedene klase *Rectangle* i *Triangle* koje vraćaju površinu pravougaonika i trougla pomoću javne funkcije *int Povrsina (void)*.

U glavnom programu:

- deklarirati objekte *rect* klase *Rectangle* i *trgl* klase *Triangle*, dodeliti im vrednosti 4 i 5 i na ekranu prikazati izračunate površine.



Zadatak 3

Povrsina pravougaonika: 20
Povrsina trougla: 10



Rešenje zadatka 3 (1)

```
#include <iostream>
```

```
using namespace std;
```

```
class Polygon {  
    protected:  
        int sirina, visina;  
    public:  
        void Postavi(int a,int b){ sirina=a; visina=b; }  
};
```

```
class Rectangle:public Polygon {           //klasa naslednik:public osnovna klasa  
    public:  
        int Povrsina (void){ return (sirina * visina); }  
};
```

```
class Triangle: public Polygon {           //klasa naslednik:public osnovna klasa  
    public:  
        int Povrsina (void){ return (sirina * visina/2); }  
};
```

```
int main() {  
    Rectangle rect;           //instanca, novi objekat  
    Triangle trgl;           //instanca, novi objekat  
    rect.Postavi(4,5);  
    trgl.Postavi(4,5);  
    cout << "Povrsina pravougaonika: " << rect.Povrsina() << endl;  
    cout << "Povrsina trougla: " << trgl.Povrsina() << endl;  
    return 0;  
}
```



Kraj prezentacije

HVALA NA PAŽNJI!

